

2010

# LEARNING IN OPEN-SOURCE SOFTWARE (OSS) DEVELOPMENT: HOW ORGANIZATIONAL AND NATIONAL CULTURE IMPACT DEVELOPERS' LEARNING

E. Ilana Diamant

*University of Pittsburgh, eid3@pitt.edu*

Sherae Daniel

*University of Pittsburgh, daniesr@ucmail.uc.edu*

Follow this and additional works at: [http://aisel.aisnet.org/icis2010\\_submissions](http://aisel.aisnet.org/icis2010_submissions)

---

## Recommended Citation

Diamant, E. Ilana and Daniel, Sherah, "LEARNING IN OPEN-SOURCE SOFTWARE (OSS) DEVELOPMENT: HOW ORGANIZATIONAL AND NATIONAL CULTURE IMPACT DEVELOPERS' LEARNING" (2010). *ICIS 2010 Proceedings*. 66.  
[http://aisel.aisnet.org/icis2010\\_submissions/66](http://aisel.aisnet.org/icis2010_submissions/66)

This material is brought to you by the International Conference on Information Systems (ICIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ICIS 2010 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact [elibrary@aisnet.org](mailto:elibrary@aisnet.org).

# LEARNING IN OPEN-SOURCE SOFTWARE (OSS) DEVELOPMENT: HOW ORGANIZATIONAL AND NATIONAL CULTURE IMPACT DEVELOPERS' LEARNING

*Research-in-Progress*

**E. Ilana Diamant**

282 Mervis Hall

Katz Graduate School of Business

University of Pittsburgh

eid3@pitt.edu

**Sherae Daniel**

244 Mervis Hall

Katz Graduate School of Business

University of Pittsburgh

sldaniel@katz.pitt.edu

## **Abstract**

*Participants in an OSS/FLOSS development project often span national and organizational boundaries, as developers from different countries and corporations join the project. A project team's national and organizational culture creates opportunities for learning from others, but may also lead to conflict and inhibit learning. This research examines developers' learning in an OSS project, and the cultural context in which learning takes place. We focus on single- and double-loop learning and examine the impact of the team's national and organizational culture on a developer's learning. Archival and survey data are collected from two large-scale Sourceforge projects. This research can contribute to the OSS literature by examining the impact of team interactions on developers' learning. Practically, administrators and managers stand to gain insight into the learning benefits of participation in OSS projects, and thus better assess their value as a training ground for global software development.*

**Keywords:** Open source, culture, learning, software development

## **Introduction**

The Global collaboration is a predominant mode of software development (Avgerou, 2002; Herbsleb, 2007). Off-shoring, outsourcing, and open sourcing are all increasing in popularity (Agerfalk and Fitzgerald, 2008). At the same time, commercial software development firms are increasingly capitalizing on the potential of Open Source Software (OSS) not only as a product but also as an informal learning environment. Informal learning environments can complement and/or reduce other educational investments for software developers such as in-house training interventions. For example, companies such as IBM, Nokia, Phillips, Apple, and Oracle have adopted OSS products and development practices and are leveraging the learning opportunities and cost savings associated with it (Wheeler, 2007; Jaaksi, 2006; Bulkeley, 2003). Major OSS systems, such as the Linux operating system (Fosfuri et al, 2008), and the Firefox browser (Fielding, 2005) are both being developed by a mix of volunteer and paid developers. Consequently, software developers must be able to interact with a diverse set of collaborators from different organizations and different places including India, China, Ireland, and the United States to remain competitive in the global marketplace. For developers seeking to gain experience in global software development, OSS projects can, because of their open, voluntary participation structure, offer valuable learning opportunities (Kang and Hahn, 2009; Singh, et al, 2010). In fact, opportunities for learning through participation in OSS projects are a major reason that developers decide to join such projects (Hertel, 2003; von Krogh et al, 2003). Nevertheless, research observations on learning in OSS report mixed results: Singh (2010) shows that developers learn through interactions with co-developers on project forums, but in Shah's (2006) interviews developers report little learning to be taking place through their participation in OSS. The practical significance, and the theoretical gap in the literature suggest that learning in OSS is not well understood in theory, despite corporate interest in OSS as an informal learning place.

Learning is primarily a social process that requires interaction with others (Bandura, 1977). Because OSS projects are very different from each other, learning opportunities are likely to vary across them (Bein and Jeffery, 2010; Capra et al, 2008). Also, the OSS developer interaction patterns, in terms of the type and frequency of contributions, vary across developers (Fang and Neufeld, 2009). This suggests that all developers may not have the same learning opportunities even when working on the same project. While there are indications from organizations and developers that OSS development offers valuable learning opportunities, there are still unanswered questions surrounding learning in OSS projects (Hertel, 2003).

The goal of this research is to examine how developers' learning is affected by their interactions on OSS projects with co-developers from various organizations and national cultures. We focus on two kinds of learning, reinforcement (single-loop) learning, and generative (double-loop) learning, which have been shown to be common in collaborative contexts within and across teams (Argyris and Schon, 1978). Because of the creative and problem-solving aspect of software development tasks, both single- and double-loop learning, as we will discuss, are crucial skills that can be practiced and refined through team interactions. The research question we pose is:

*RQ: How does OSS team interaction affect a developer's single- and double-loop learning?*

## **Background**

### ***Single- and Double-Loop Learning***

Effective learning depends on opportunities to interact with, or observe, one's peers (Bandura, 1977; Webb, 1989). Immersive participation in a task, in particular, that allows learning by doing (Schank, 1999). In online environments situated learning allows for both immersive and peripheral participation of peer activities (Lave and Wenger, 1991) which facilitates observational learning. Situated online learning in OSS is facilitated by the asynchronous nature and the open boundaries of the OSS project itself: as new participants join the project team at different points in time, they can practice or learn new skills through participation or observation (lurking). At the same time, the existing team members stand to learn from the contributions of newcomers, especially if newcomers

have a different background and are able to contribute novel perspectives on the task (Levine et al, 2003; Nemeth, 1986; von Krogh et al, 2003).

Following Argyris and Schon's (1978) original formulation, single-loop learning is defined as the ability to master a skill through repetitive practice that reinforces an already acquired behavior (Argyris and Schon, 1978). Single-loop learning occurs when an individual (or group) shows incremental improvements in their performance, practice, or other work-related behavior, as measured by some fixed standard of performance. Double loop learning works through reflection on one's work routines and habitual practices, such as the way the individual learner or a team approaches a problem, their process for identifying alternative solutions, evaluating them and deciding on a course of action (Argyris and Schon, 1978). Psychologically, double-loop learning consists of awareness about one's behavior while that person or team is engaged in various tasks that have learning value to them. Socially, double-loop learning is effectively practiced in team settings or, at least, minimal collaborative settings (e.g., paired programming) because the perspectives, problem-solving tactics, and ideas of others can help the learner contrast their own learning routines and tactics with those of their collaborators.

Organizations with relatively flat hierarchies and self-managed teams are more conducive to double-loop learning because they encourage employee initiative, risk-taking, and discretion in which tasks and projects to work on (Yeomans, 2000). It might seem that OSS projects are a context that would encourage double-loop learning because of the self-managed teams and discretion that developers have over their contributions. However, OSS developers are often employed by hierarchical organizations and are assigned to work on commercial software development projects; the practices and habits that one develops in their daytime work-context might transfer to one's OSS work behavior, as individuals differ in their cognitive ability to switch gears across different contexts (Louis and Sutton, 1991). For those reasons, we do not assume that double-loop learning is the only kind of learning occurring in OSS projects, but rather, we examine the possibility that both single- and double-loop learning might coexist as a result of the developer's multiple work roles, habitual practices, and cognitive demands on their attention.

### ***Impacts of Culture(s) on Learning***

In global software development teams, such as OSS projects, there are likely to be multiple constructions of culture (Eisenberg and Riley, 2001). We examine two aspects of culture that are common among OSS developers. Following the approach of Gibbs (2009) we conceptualized team culture as a dynamic combination of members' national and organizational cultures. Organizational culture is generally defined as the set of shared assumptions, values, and behaviors among members of an organization (Schein, 1992). National culture is similarly defined as the set of assumptions, values and behaviors that are shared among members of an ethnicity (e.g., Hofstede, 1980). Below we describe the characteristics of each kind of culture and hypothesize how they are likely to affect a developer's learning.

### ***National Culture***

Cultural homogeneity prompts validation of behaviors and views that one has already adopted (Nemeth, 1986). When exposed to similar viewpoints, team communication serves to confirm and reinforce what a person already knows. At From the perspective of an individual developer, interactions with culturally similar co-developers are likely to reinforce the skills one already has or the problem-solving approach that one is already following. We expect that the learning that takes places through developers' interactions with others from similar cultures will be a reinforcement type of learning, known as single-loop learning. For a developer with strong programming skills, the opportunity to practice them in a new context will be beneficial. Similarly, a developer that is a non-expert can strengthen her skills through her work on the project and interactions with expert co-developers.

*Hypothesis 1: A developer's interactions with culturally similar developers on a project will promote the acquisition of single-loop learning skills; the higher the number of culturally-similar co-developers that a developer interacts with, the greater the likelihood that he/she will acquire single-loop learning skills.*

Teams that include members from different national backgrounds offer learning opportunities from interactions and observation of others' behavior and problem-solving activities (Williams and O'Reilly, 1998, Nemeth, 1986). Cross cultural teams, as mentioned above, offer opportunities for members to be exposed to different viewpoints and problem-solving approaches (Nemeth, 1986). When exposure to diverse information or knowledge takes place, that is, when members have the opportunities to interact with others who do not share the same cultural background, then generative, or double-loop learning is more likely to occur (Peltier, 2005; Argyris and Schon, 1978). Double loop learning rests on a reflective awareness and re-examination of one's skills or viewpoints. In OSS projects, double loop learning might involve rethinking the way a coding problem is addressed, or reframing a purely technical problem as a different kind of problem, such as a usability one. More generally, double loop learning in an OSS project is when a developer reconsiders the overall problem-solving approach that a he takes when working alone or with others. Developers from different national cultures are likely to bring a variety of creative thinking and problem-solving to a project, and will likely have had experiences with distinct development methods. To benefit from that diversity, however, one must devote considerable attention and time to communicating with one's co-developers on the project. Exposure to too little or too much diversity of viewpoints and development techniques is less likely to be a learning experience; with a moderate level of diversity, team members are more likely to attend to and use the information and expertise that others bring to the task (Dahlin et al, 2005). For that reason, we expect that developers who interact with a moderately diverse group of co-developers to benefit most from that diversity, that is, to engage in reflective, generative learning (double-loop learning).

*Hypothesis 2: There is an inverted U-shaped curvilinear relationship between the national cultural diversity of a focal developer's co-developers on a project and the focal developer's double-loop learning. Interactions with a moderate number of culturally dissimilar developers will have an amplifying effect, while interactions with a low and a high number of dissimilar developers will have a dampening effect on that developer's acquisition of double-loop learning.*

### **Organizational Culture**

As commercial software development firms become involved in OSS development or adopt OSS products and practices into their business models, an increasing number of commercially employed developers are joining OSS projects (Krishnamurthy, 2005). The norms, values and practices of a developer's employing organization might not always be consistent with the distinct ideology and practices of the OSS movement (Hertel et al, 2003; Stewart and Gosain, 2006). Organizational culture may affect learning in the OSS context in two ways: first, the focal developer's employing organization might have a corporate culture that is supportive (or unsupportive) of OSS development. Organizational support for OSS development and developers can take on various forms. Firms often adopt an OSS application, and market and advertise it as part of their product mix. Attention drawn to that product increases participation as more developers discover the original OSS project and decide to join (Hars & Ou, 2002). Employees of software development firms may also receive tangible or intangible benefits for participating in OSS, such as flexible time, increased pay, or the opportunity to work on projects of their discretion which is not always the case in commercial development. Software firms also increasingly consider OSS development as an advantage in hiring or promotions (Krishnamurthy, 2005). Less obvious support for OSS development can be seen in the organization's norms, routines, and practices, and specifically those pertaining to software development. OSS development is characterized by norms distinct from commercial development (Raymond citation); the extent to which an organization adopts any of those norms or supports them among its developers, that can be considered as evidence of an OSS-supportive culture.

An OSS-supportive culture, we argue, is more likely to facilitate double-loop learning because of the additional motivation and incentives that are available to developers employed by that organization. Whereas developers join OSS projects for a variety of reasons (Lakhani and von Hippel, 2003), the additional intangible and tangible incentives from an OSS supportive organization will function as motivation for them to continue participating or keep joining new projects. Oftentimes developers from a supportive corporate environment view their OSS participation as a part of their job or in a more positive light than their day-time job. Moreover, firms that support the OSS movement in general as opposed to a single OSS application have a more beneficial effect on OSS participation patterns overall (Wagstrom et al, 2010). A similar effect, we argue might be present on the developer's double loop learning, leading to increased effort and willingness to approach OSS as a generative learning opportunity and engage in the reflection-adaptation process that is central to double-loop learning.

*Hypothesis 3a: Developers whose employing organization has an OSS-supportive culture will be more likely to engage in double-loop learning.*

Organizational culture may also affect learning depending on the developer's interactions and communication with others on the OSS project. A developer might interact with co-developers that are employed by companies that vary in their supportiveness to OSS development practices. Organizational support for OSS development practices, and developers, implies that experimentation with different development practices is encouraged; a culture of experimentation with respect to software development is more likely to facilitate double-loop learning, according to Argyris and Schon's (1978) model of the learning organization. A culture of experimentation also implies that developers who decide to experiment with OSS practices are motivated by goals of learning rather than by goals of performance (e.g., Cacioppo and Petty, 1982). From the perspective of a developer who interacts with other co-developers that are employed by supportive organizations, their learning motives are important because a similarity of motivation and goals (learning goals) promotes effective communication in teams (Crown and Rosse, 1995; Weingart and Weldon, 1991) which in turn is crucial for double-loop learning (Argyris and Schon, 1978). Having similar learning goals also promotes openness to the OSS values that are not common in commercial development organizations, such as collective credit, open team boundaries, and early iterative code releases. Thus, interacting with developers who are employed by an OSS-supportive organization will promote double loop learning because of the likely congruence of motivations and values that developers might share towards OSS.

*Hypothesis 3b: A developer's interactions with developers from OSS-supportive organizational cultures will promote the acquisition of double-loop learning skills; the higher the number of co-developers from OSS-supportive organizational cultures that a developer interacts with, the greater the likelihood that he/she will engage in double-loop learning.*

### ***Interactive Effect of Organizational and National Culture***

Team culture, as mentioned earlier, is a dynamic construct, evolving from of members' national cultures and their organization's culture (Gibbs, 2009). For that reason we consider both their direct and interactive effects on learning. As hypothesized earlier, organizational culture will likely affect a developer's learning depending on the organizational culture of the other members in the OSS project with whom the focal developer interacts. While an OSS-supportive culture is likely to promote double loop learning, through interactions with developers that are part of such a culture in their workplace, that effect can be further qualified by considering the developers' national culture. The learning benefits of an OSS-supportive culture are likely to be amplified or dampened depending on whether the participants a developer interacts with share the same or different national culture. Interacting with culturally diverse developers is likely to offer greater opportunities for learning, in addition to those offered by an OSS supportive culture in the workplace. We expect that a developer will have greater opportunity for reflection on their coding practices, and therefore will experience greater double-loop learning, when he/she interacts with developers from different national cultures, because such diversity is likely to expose one to different work styles, problem-solving approaches, and development techniques..

*Hypothesis 4: A developer's interactions with developers from OSS-supportive organizational cultures will promote the acquisition of double-loop learning depending on the developers' national culture: interactions with culturally dissimilar developers will have an amplifying effect, while interactions with dissimilar developers will have a dampening effect on that developer's double-loop learning.*

## **Methodology**

### ***Sample***

We are using a combination of survey and archival data. The archival data is publicly available on Sourceforge.net and has been used in prior studies of OSS development (e.g., Grewal et al, 2006). A survey of OSS developers will be conducted to assess the dependent variables, that is, the extent to which learning occurs across OSS projects. We

sample developers from two large projects registered on SourceForge, Mumble and Pidgin. Sampling developers from two projects helps alleviate potential bias associated with sampling participants from only one project. Using a single project as the only source of data might lead to responses and findings that are highly idiosyncratic for that project and its developers. With a second project as a data source that bias is less likely to occur without at the same time introducing too much exogenous variation in the responses. Sampling developers from two projects minimizes the potentially different effect of the project's work environment on learning. Work environment includes things such as the mode of coordination of the software development process, responding to user feedback, managing the release process.

To select these two projects we use several criteria that facilitate testing our model. Both projects are currently active, and use Sourceforge forums to coordinate the software development. Active forums throughout their course of development allow the capture of a large number of developer communications. Both use object-oriented programming languages and develop instant messengers, which minimizes the likelihood that developers' learning ability is affected differently by the nature of the technology or programming language of their project. Controlling for developers' prior experience and expertise level, the effect of the technology will likely be similar across individuals. Finally, both have an international group of developers that contribute to them which allows us to examine geographic dispersion.

### Measures

Developer interactions in this study are operationalized as the number of co-developers in an OSS project with whom the focal developer (survey respondent) has communicated in the project's forums (Table 1). That data is captured in the Sourceforge archives. The developers' national culture diversity of will be gathered from archival data and assessed with Blau's (1977) diversity index. Organizational support for OSS will be assessed with survey items. Single- and double-loop learning will be assessed with questionnaires previously used in the literature which we adapted for the context of OSS projects. Learning is assessed as the developer's "perceived" learning. The acquisition or development of single-loop learning reflects the degree to which the developer thinks that his/her participation in a given project has increased his/her mastery of already acquired skills. The acquisition of double-loop learning reflects the degree to which the developer thinks that his/her participation in a given project has increased his/her acquisition or development of single – or double-loop learning. Sample items for assessing double-loop learning include: "Participating in this project has helped me understand how I think when I try to solve a problem/made me aware of the thinking process/style I follow when I approach a new problem/improved my ability to learn from mistakes, errors or problems that I encountered when developing this software. Sample items for assessing single-loop learning include: "participating in this project helped me learn how to apply what I already know about programming"(Table 1).

Learning orientation, and need for cognition, which reflect individual differences in learning dispositions, are assessed as controls with scales in the literature: Gray and Meister (2004) and Cacioppo and Petty (1982) respectively. Demographic information and the developers' level of expertise and motivation for joining their OSS project are also assessed in the survey, the latter with open-ended questions.

**Table 1. Constructs and Measures**

Construct	Measure
Developer Interactions with others from a similar national culture	Archival data: co-developers' location/country. Number of co-developers that the focal developers interacts with in an OSS project
Developer Interactions with others from a dissimilar national culture	Archival data: co-developers' location/country. Blau's diversity index.

Developer Interactions with others from an OSS (un)supportive organizational culture	Survey items, e.g., “My organization is (un)supportive of OSS development”, “we are encouraged to join OSS projects”.
Single-loop Learning	Survey items, e.g., “Participating in this project helped me learn how to apply what I already know about programming”.
Double-loop Learning	Survey items, e.g., “Participating in this project helped me rethink the way I approach a coding problem”.
Developer Demographics & Expertise	Survey items (e.g., organizational affiliation, time on Sourceforge, reason/motivation for joining project, learning orientation).

### Data Analysis

The proposed hypotheses will be tested with OLS regression. Survey responses will be matched with the developer's interaction patterns in the Sourceforge project forums. Some descriptive statistics on the projects are presented in Table 2.

Table 2. OSS Project Characteristics		
	<i>Pidgin</i>	<i>Mumble</i>
Number of forum posters	5,564	790
Number of forum posts	21,459	4,253
International team composition	Yes	Yes
OO programming	Yes	Yes
Registration Date	2003	2005
Number of posts by person with most posts	1219	662

### Contributions

This study is designed to make a contribution to the OSS literature by drawing on findings from the learning literature. In addition, findings of this study may also add to our general understanding of how cultural and organizational differences may affect learning in teams that are socio-technically distributed. That is, teams that are working on a creative problem-solving task or project that is divided into subcomponents, analogous to software modules and components, which are completed by different members in an asynchronous fashion. How learning occurs in such distributed work contexts isn't well understood in the current state of the learning literature (Vatrapu, 2008), and findings from this study stand to contribute to that line of research. In addition, we consider the effects of team diversity as they are represented by actual member-to-member interactions rather than as a team-level demographic variable. This view of communication is an important point in examining the benefits of diversity, because diversity research has tended to assess the impact of team diversity as a team-level variable without looking into the internal communication practices among members, that is without examining whether the members of diverse teams communicate with each other, and more importantly, with culturally dissimilar colleagues (Gallivan and Srite, 2005; Shachaf, 2008; Williams and O'Reilly, 1998; Zhang et al, 2007).



## Conclusion

The prevalence of off-shoring and open-source collaborations, as well as corporate involvement in OSS development suggest that global software development is becoming a dominant mode of IS development. The value of global IS projects, and OSS in particular, for developers and their employing organizations needs to be examined from multiple aspects beyond the performance and success dimensions. Opportunities for informal learning, training, and skill development are an important and potentially value-added aspect of global IS projects such as OSS ones, however the benefits and challenges associated with those opportunities are not well understood. This study examines the learning benefits of OSS projects while considering the potential challenges that their internal diversity may pose. As OSS projects become more diverse including participants from different organizations and cultures, learning opportunities multiply but so do their challenges. Understanding such learning opportunities and the risk associated with internal team communication patterns stands to benefit project sponsors, managers and the developers themselves.

## References

- Agerfalk, P and Fitzgerald, B. "Outsourcing to an Unknown Workforce: Exploring Opensourcing as a Global Sourcing Strategy", *MIS Quarterly* (32:2), 2008, pp. 385-410
- Argyris, C. and Schön, D. A. *Organizational Learning*. Reading, Mass., Addison-Wesley Publishing. 1978
- Avgerou, C *Information Systems and Global Diversity*. Oxford/New York: Oxford University Press, 2002
- Bandura, A. *Social Learning Theory*. Prentice-Hall, Englewood Cliffs, NJ. 1977
- Barki, H. and Hartwick, J. "Interpersonal Conflict and its Management in Information System Development," *MIS Quarterly*, 2001, (25: 2) pp.195
- Bein, W., and Jeffery, C. "Towards an Openness Rating System for Open Source Software," *Hawaii International Conference on System Sciences*, 2010
- Blau, P. 1977. *Inequality and Heterogeneity*. New York: Free Press
- Brookfield, S. D. "The Concept of Critically Reflective Practice", in A. L. Wilson & E. R. Hayes (Eds.), *Handbook of Adult and Continuing Education* (pp. 33-49). San Francisco, CA: Jossey-Bass, 2000
- Bulkeley, W. M. "Free Software—Out of the Shadows," *The Wall Street Journal, Technology (A Special Report)*, March 31, 2003, p. R6
- Cacioppo, J.T. and Petty, R. E. "The Need for Cognition," *Journal of Personality and Social Psychology*, (42:1), 1982, pp. 116-131
- Capra, E., Francalanci, C., and Merlo, F. "An Empirical Study on the Relationship among Software Design Quality, Development Effort, and Governance in Open Source Projects," *IEEE Transactions on Software Engineering* (34:6) 2008
- Crown, D. F., and Rosse, J. G. "Yours, Mine, and ours: Facilitating Group Productivity through the Integration of Individual and Group Goals. *Organizational Behavior and Human Decision Processes*, 64, 1995, pp. 138–150

- Dahlin, K. B., Weingart, L. R., and Hinds, P. "Team Diversity and Information Use". *Academy of Management Journal*, 48, 2005, pp.1107-1123
- Eisenberg, E., and Riley, P. "Organizational Culture". In F. M. Jablin & L. L. Putnam (Eds.), *The New Handbook of Organizational Communication*, pp. 291- 322. Thousand Oaks, CA: Sage Publications.2001
- Fang, Y. and Neufeld, D. "Understanding Sustained Participation in Open Source Software Projects". *Journal of Management Information Systems*, (25:4), 2009, pp. 9-50
- Fielding, R. T. "Software Architecture in an Open Source World. In *27th International Conference on Software Engineering*, 43, Presented at the 2005 *International Conference on Software Engineering (ICSE 2005)*, St. Louis, MO, USA, 2005
- Fosfuri, A., Giarratana, M., S., and Luzzi, A. "The Penguin Has Entered the Building: The Commercialization of Open Source Software Products", *Organization Science*, (9:2), pp. 292–305
- Foldy, E. G., and Creed, W. E. D. "Action Learning, Fragmentation, and the Interaction of Single-, Double- and Triple-Loop Learning". *Journal of Applied Behavioral Science*, 35, 1999, pp. 207-227
- Gallivan, M., and Srite, M. "Information Technology and Culture: Identifying Fragmentary and Holistic Perspectives of Culture, *Information and Organization*, (15:4), 2005, pp.295-338
- Gibbs, G. "Learning how to Learn Using a Virtual Learning Environment for Philosophy', *Journal of Computer Assisted Learning*, 15, 1999, pp.221-31
- Gibbs, J. L. "Culture as Kaleidoscope: Navigating Cultural Tensions in Global Collaboration". In *Proceeding of the 2009 International Workshop on Intercultural Collaboration* (pp. 89--98). NY: ACM.2009
- Gray, P.H. and D.B. Meister, D.B. "Knowledge Sourcing Effectiveness". *Management Science* (50:6), 2004, pp. 821–834
- Grewal, R., Lilien, G.L. and Mallapragada, G. "Location, Location, Location: How Network Embeddedness Affects Project Success in Open Source Systems", *Management Science* (52:7), 2006, pp. 1043–1056
- Hars, A., and Ou, S. "Working for Free? Motivations for Participating in Open-Source Projects". *International Journal of Electronic Commerce*, (6:3), 2002, pp. 25-39
- Henkel, J. "Champions of Revealing - The Role of Open Source in Commercial Firms" *Industrial and Corporate Change*, (18:3), 2006, ,pp. 435–471
- Herbsleb, J. "Global Software Engineering: The Future of Socio-Technical Coordination", in *Proceedings of the International Conference on Software Engineering*, 2007, pp 188-198
- Herbsleb, J. D., Mockus, A., Finholt, T. A., & Grinter, R. E. "Distance, Dependencies, and Delay in a Global Collaboration", in *Proceedings, ACM Conference in Computer-Supported Cooperative Work*, Philadelphia, PA, 2000
- Hertel, G., Niedner, S., and Herrmann, S. "Motivation of Software Developers in Open Source Projects: an Internet-based Survey of Contributors to the Linux Kernel," *Research Policy* (32) 2003, pp. 1159-1177.
- Hofstede, G. 1983. "The Cultural Relativity of Organizational Practices and Theories". *Journal of International Business Studies*, (14:2), 1983, pp. 75–89.
- Jaaksi, A. (2006), 'Building Consumer Products with Open Source Communities – the Maemo and 770 Experiences,' Presentation at LinuxWorld 2006, Boston,  
[http://www.kotiposti.net/jaaksi/ME9\\_LinuxWorld\\_2006\\_AriJaaksi\\_.pdf](http://www.kotiposti.net/jaaksi/ME9_LinuxWorld_2006_AriJaaksi_.pdf).

- Janis, I. L. *Victims of Groupthink* (2nd ed.). Boston: Houghton Mifflin. 1982
- Jehn, K. A. "A Multimethod Examination of the Benefits and Detriments of Intragroup Conflict." *Administrative Science Quarterly*, 40, 1995, pp. 256-282
- Kanfer, R. (1990). "Motivation Theory and Industrial and Organizational Psychology". M. Dunnette, L. Hough, ed. *Handbook of Industrial and Organizational Psychology. Consulting Psychology Press*, 2<sup>nd</sup> edition, Palo Alto, CA, pp. 75–170
- Kang, K. and Hahn, J. "Learning and Forgetting Curves in Software Development: Does Type of Knowledge Matter?". *International Conference on Information Systems*, 2009, Phoenix, AZ, US
- Krishnamurthy, S.. "An Analysis of Open Source Business Models". In J. Feller, B. Fitzgerald, S. Hissam, & K. R. Lakhani (Eds.), *Perspectives on Free and Open Source Software*, Cambridge, MA: MIT Press, 2005
- Lakhani, K., and von Hippel, E. "How Open Source Software Works: 'Free' User-to-User Assistance," *Research Policy* (32:6), 2003, pp. 923-943
- Lave, J., and Wenger, E. *Situated Learning: Legitimate Peripheral Participation*, Cambridge University Press, Cambridge, UK, 1991
- Levine, J. M., Choi, H-S., & Moreland, R. L. (2003). "Newcomer Innovation in Work Teams". In P. Paulus & B. Nijstad (Eds.), *Group Creativity: Innovation Through Collaboration*, (New York, 2003, pp. 202-224
- Li, M, Lin, Z., and Xia, M. "Leveraging the Open Source Software Movement for Development of China's Software Industry, *Information Technology and International Development* (2:2), 2004, pp. 45–63
- Louis, M.R. and Sutton, R.I. "Switching Cognitive Gears: From Habits of Mind to Active Thinking", *Human Relations*, 44, 1, 1991, pp.55-76
- Meyer, H. H., and Raich, M. S. "An Objective Evaluation of a Behavior Modeling Training Program. *Personnel Psychology*, 36, 1983, pp. 755–761
- Mockus, A., Herbsleb, J. D., Finholt, T.A., Grinter, R "An Empirical Study of Global Software Development: Distance and Speed", in *Proceedings of International Conference on Software Engineering*, 2001, pp.81-90
- Nemeth, C. 1986 "Differential Contributions of Majority and Minority Influence." *Psychological Review*, 93, 1986, pp. 23-32
- Peltier, J. W., Hay, A., & Drago, W. (2005). "The Reflective Learning Continuum: Reflecting on Reflection". *Journal of Marketing Education*, (27:3), 2005, pp.250-263
- Sacks, M. *On-the Job Learning in the Software Industry*. Quorum Books, 1994. Westport, CT.
- Schein, E. H. *Organizational Culture and Leadership* (2nd ed.). San Francisco: Jossey-Bass Publishers. 1992
- Senge, P.M. *The Fifth Discipline*. New York: Doubleday Currency. 1990
- Schank, R. C., Berman, T. R., and Macpherson, K. A. "Learning by Doing". In C. M. Reigeluth (ed.), *Instructional-Design Theories and Models*, (Vol. II), pp. 161–181. Mahwah, NJ: Lawrence Erlbaum Associates, Inc. 1999
- Shachaf, P. "Cultural Diversity and Information and Communication Technology Impacts on Global Virtual teams: An Exploratory Study, *Information and Management*, (45:2), 2008, pp.131-142

- Shah, S.K. 2006. Motivation, Governance and the Viability of Hybrid Forms in Open Source Software Development. *Management Science* (52:7), 2006, pp. 1000-1014
- Singh, P.V., Youn, N, and Tan, Y. "Developer Learning Dynamics in Open Source Software Projects". Forthcoming in *Information Systems Research*, 2010
- Stewart, K J., and S. Gosain. "The Impact of Ideology on Effectiveness in Open Source Software Development Teams". *MIS Quarterly* (30:2), 2006, pp. 291-314
- Subramanyam, R. and Xia, M. "Free/Libre Open Source Software Development in Developing and Developed Countries: A Conceptual Framework with an Exploratory Study". *Decision Support Systems*, (46:1), 2008, p.173
- Vatrapu, R. "Cultural Considerations in Computer Supported Collaborative Learning". *Research and Practice in Technology Enhanced Learning*, (3:2), 2008, pp.159-201
- von Krogh, G., Spaeth, S., Lakhani, K. "Community, Joining, and Specialization in Open Source Software Innovation: A Case Study". *Research Policy*, 2003
- Wagstrom, Herbsleb, Kraut, and Mockus. "The Impact of Commercial Organizations on Volunteer Participation in an Online Community". Academy of Management Annual Meeting. Montreal, Canada, August 2010
- Webb, N. M. "Peer Interaction and Learning in Small Groups". *Journal of Educational Research*, 13, 1989, pp. 21-39
- Weingart, L., and Weldon, E. (1991). Processes that Mediate the Relationship between a Group Goal and Group Member Performance. *Human Performance*, 4, 1991, pp. 33-54
- Wheeler, D. "Why Open Source Software / Free Software (OSS/FS, FLOSS, or FOSS)? Look at the Numbers", 2007. Available at: [http://www.dwheeler.com/oss\\_fs\\_why.html#market\\_share](http://www.dwheeler.com/oss_fs_why.html#market_share). (retrieved 06/22/10)
- White, B. Y., Shimoda, T. A., Frederiksen, J. R. "Enabling Students to Construct Theories of Collaborative Inquiry and Reflective Learning: Computer Support for Metacognitive Development". *International Journal of Artificial Intelligence in Education*, 1999, pp. 15-182
- Williams, K. Y., and O'Reilly, C. A. "Demography and Diversity in Organizations." In Barry M. Staw and Robert M. Sutton (eds.), *Research in Organizational Behavior*, 20, pp. 77-140. Stamford, CT: JAI Press, 1998
- Yeh, Y., and Chou, H. "Team Composition and Learning Behaviors in Cross-functional Teams". *Social Behavior and Personality*, 33, 2005, pp. 391-402
- Yeomans, L. "Does Reflective Practice Have Relevance for Innovation in Public Relations?" *Journal of Communication Management*, (5:1), 2000, pp.72-81
- Zhang , D., Lowry , P. Zhou ,L., and Fu, X. "The Impact of Individualism-Collectivism, Social Presence, and Group Diversity on Group Decision Making Under Majority Influence", *Journal of Management Information Systems*, (23:4), 2007, pp.53-80